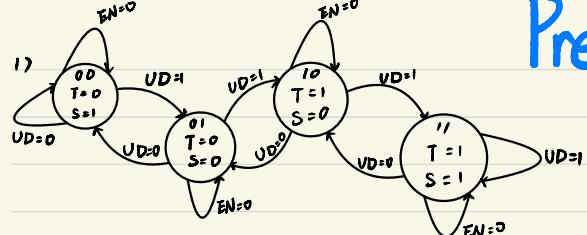


Prelab



2) & 3)

S_1	S_0	UD	EN	RS	NS_1	NS_0	Taken	Strong
0	0	0	0	0	0	0	1 ✓	0
0	0	0	1	0	0	0	1 ✓	1
0	0	1	0	0	0	0	1 ✓	2
0	0	1	1	0	0	1 ✓	1 ✓	3
0	1	0	0	0	0	1 ✓	0	4
0	1	0	1	0	0	0	0	5
0	1	1	0	0	0	1 ✓	0	6
0	1	1	1	0	1 ✓	0	0	7
1	0	0	0	0	1 ✓	0	1 ✓	8
1	0	0	1	0	0	1 ✓	1 ✓	9
1	0	1	0	0	1 ✓	0	1 ✓	10
1	0	1	1	0	1 ✓	1 ✓	0	11
1	1	0	0	0	1 ✓	1 ✓	1 ✓	12
1	1	0	1	0	1 ✓	0	1 ✓	13
1	1	1	0	0	1 ✓	1 ✓	1 ✓	14
1	1	1	1	0	1 ✓	1 ✓	1 ✓	15

For SOP expression of Next State:

$$NS_1 : \sum(I_{S1}UDEN) = m_7 + m_8 + m_{10} + m_1 + m_{12} + m_{13} + m_{14} + m_{15}$$

$$= S_1 S_0 UDEN + S_1 S_0' UDEN' + S_1 S_0' UDEN + S_1 S_0 UDEN'$$

$$+ S_1 S_0 UDEN + S_1 S_0' UDEN + S_1 S_0 UDEN$$

$$NS_0 : \sum(I_{S0}UDEN) = m_3 + m_4 + m_6 + m_9 + m_{11} + m_{12} + m_{14} + m_{15}$$

$$= S_1' S_0 UDEN + S_1' S_0' UDEN' + S_1' S_0' UDEN + S_1' S_0 UDEN'$$

$$+ S_1' S_0 UDEN + S_1' S_0' UDEN + S_1 S_0 UDEN$$

Taken = S_1

$$\text{Strong} = S_1 \oplus S_0 = S_1 S_0 + S_1' S_0'$$

EN	$S_1 S_0$	S_0			
		00	01	11	10
UDEN	00	m ₀ 0000	m ₄ 0100	m ₁₀ 1100	m ₈ 1000
	01	m ₁ 0001	m ₅ 0101	m ₁₃ 1101	m ₉ 1001
	11	m ₃ 0011	m ₇ 0111	m ₁₅ 1111	m ₁₁ 1011
	10	m ₂ 0010	m ₆ 0110	m ₁₄ 1110	m ₁₀ 1010

NS_1

$$S_0 S_1 \oplus S_0 UD \oplus S_0 UDEN \oplus S_0 UDEN' \quad | \quad S_0 EN' \oplus S_0 S_1 EN \oplus S_0 UDEN \oplus S_0 UDEN$$

$$NS_1 = S_0 S_1 + S_0 UD + S_0 UDEN + S_0 UDEN' \quad | \quad NS_0 = S_0 EN' + S_0 S_1 EN + S_0 UDEN + S_0 UDEN$$

EN	$S_1 S_0$	S_0			
		00	01	11	10
UDEN	00	m ₀ 0000	m ₄ 0100	m ₁₀ 1100	m ₈ 1000
	01	m ₁ 0001	m ₅ 0101	m ₁₃ 1101	m ₉ 1001
	11	m ₃ 0011	m ₇ 0111	m ₁₅ 1111	m ₁₁ 1011
	10	m ₂ 0010	m ₆ 0110	m ₁₄ 1110	m ₁₀ 1010

NS_0

Post lab

2. Your "Verilog version II" code was likely longer than the version I code. Why don't we generally design state machines the way we did in version I? Will the version I way generally be shorter? (5 points)

Although Version 2 is longer than Version 1, but for the transmission equation in Version 1 about N_{S0} and N_S , you need to first write truth table, then use canonical SOP to express function and then use K-map to minimize it which more complex than Version 2. Also which version is shorter depend on the specific situations.

3. Say we wanted to change the design so that the counter wraps around rather than saturates. Which of your two designs would be easiest to change? Explain why. (5 points)

It must be Version 2. Because if you want the counter wraps around rather than saturates, you only need to change the if else content in zero and three state S_0 so that $N_S = 0$ instead of three $N_S = 3$ instead of zero respectively. However for Version 1, it will become harder, since you need to create the new truth table, then find the new minterm that can make the switch function to be one and then minimize them.

4. Provide another version of your next state logic for "Verilog version II", but this time using + and - operators. Assume the state assignment is the obvious two-bit assignment ($ZERO=2'b00$, $ONE=2'b01$, etc.). It does not have to compile. (5 points).

if $|S==2'b00 \& S==2'b11|$

$ns = S_1$

else if $(UD \& en)$

$ns = S_1$

else if $(\neg ud \& \neg en)$

$ns = S_1$

else

$ns = S_0$